A Simplified Graphical Representation of the Holographic Principle

Douglas C. Youvan

doug@youvan.com

September 21, 2023

The Holographic Principle posits that the information contained within a given volume of space can be represented on the boundary of that space. It's a profound idea that challenges our understanding of space, gravity, and information. To visually capture this concept, we've designed a simple, interactive 3D representation. https://www.youtube.com/watch?v=RSIB-Ji0LQo

The Setup:

- **The Sphere:** Our chosen space is a transparent 3D sphere. The interior of this sphere represents the volume of a region in space, while its surface embodies the boundary on which information can be encoded.
- **Internal Points (Red Balls):** Inside the sphere are randomly positioned red balls, representing dynamic information or events happening within the space's volume. Their positions and interactions are akin to the complexities of quantum field theories inside the volume.
- **Boundary Point (Colored Spot):** The vividly colored moving spot on the sphere's surface represents the encoding of the volume's internal information on the boundary. Its color and movement change in response to the positions and dynamics of the red balls inside the sphere, illustrating the idea that all the information inside can be "holographically" represented on the boundary.

Interpretation:

When the red balls inside the sphere change position, the color and movement of the surface spot change correspondingly. This simultaneous change exemplifies the core idea of the Holographic Principle: the entirety of the information within the volume (the changing positions of the balls) can be fully encoded and represented on just the boundary of that volume (the color and movement of the spot).

One minute video:

https://www.youtube.com/watch?v=RSIB-Ji0LQo

Code written by GPT-4:

mathematica Copy code

```
Animate[
 Module[{pts = RandomPoint[Ball[], 3], avg, rgb},
  avg = Mean[pts];

  (* Determine the dominant RGB channel *)
  With[{maxDim = Ordering[Abs[avg], -1][[1]]},
   rgb = Switch[maxDim,
     1, {1, Abs[avg[[2]]], Abs[avg[[3]]]},
     2, {Abs[avg[[1]]], 1, Abs[avg[[3]]]},
     3, {Abs[avg[[1]]], Abs[avg[[2]]], 1}
     ]
   ];

  Graphics3D[
   {
    Opacity[0.3], Sphere[],
    Darker[Red, 0.3], PointSize[0.03], Point[pts],
    RGBColor[rgb], PointSize[0.05],
    Point[{Cos[t] Sin[2 t], Sin[t] Sin[2 t], Cos[2 t]}]
    },
   Boxed -> False,
   Lighting -> "Neutral"
   ]
  ],
 {t, 0, 20 Pi, 0.1}
]
```

```
Animate[
 Module[{pts = RandomPoint[Ball[], 3], avg, rgb},
  avg = Mean[pts];

  (* Determine the dominant RGB channel *)
  With[{maxDim = Ordering[Abs[avg], -1][[1]]},
   rgb = Switch[maxDim,
     1, {1, Abs[avg[[2]]], Abs[avg[[3]]]},
     2, {Abs[avg[[1]]], 1, Abs[avg[[3]]]},
     3, {Abs[avg[[1]]], Abs[avg[[2]]], 1}
     ]
   ];

  Graphics3D[
   {
    Opacity[0.3], Sphere[],
    Darker[Red, 0.3], PointSize[0.03], Point[pts],
    RGBColor[rgb], PointSize[0.05],
    Point[{Cos[t] Sin[2 t], Sin[t] Sin[2 t], Cos[2 t]}]
    },
   Boxed -> False,
   Lighting -> "Neutral"
   ]
  ],
 {t, 0, 20 Pi, 0.1}
]
```